Model-based Development of Graphical User Interfaces[1]

Dániel Tóth and Eszter Jósvai

The development of computer technology vastly broadened the range of potential users. Previously the primary users of computer applications were developers themselves, nowadays most applications are developed for end users who use them for regular work and have no special understanding of the inner workings of the system. These users need graphical user interfaces (GUIs) designed with these requirements in mind. Another important trend in application design is the requirement to be able to run on a wide range of different platforms.

Present day GUI designer tools are very advanced in that they make the design process a very simple task. They usually feature code generation facilities to remove the requirement of lengthy coding. However they are typically tied to a single programming language and platform. There are attempts to create platform-independent GUI description techniques, but they typically approach the problem by adding another layer above existing frameworks and widget libraries. Usually some interpreter is used to translate the platform-independent description to the native widget set in run-time.

True platform independence can be achieved trough the use of OMG's Model Driven Architecture (MDA) guidelines. The MDA defines a development procedure where the application is designed as a platform-independent model which is automatically transformed into platform specific models which serve as a base for automatic code generation for a wide variety of platforms. The ability to modify the application model easily, then propagate the modification automatically to the generated appliation is particularly important in GUI design, since it requires frequent conciliation with the end users.

We created a GUI modelling system using the VIATRA2 transformation and modelling framework that allows the generation of native code for different platforms. It can also take the platform independent GUI model from different kinds of model sources. Our prototype implementation used UML models for input because it is an industry standard for software design and documentation. UML by itself is not useable for GUI modelling and this has been a major obstacle in utilizing the MDA concept for such development. We created a UML profile to make platform independent GUI modelling in UML possible.

We have found that user interface design is not as easy in UML as in the special GUI design tools. Therefore we conducted further research in the field of domain specific modelling (DSM). DSM allows the creation of an easy to use visual GUI model editor, similar to the plaform specific design tools, while maintaining the platform independence and the ability to output UML models for unified documentation of the application.

References

[1] D. Varró and A. Pataricza. VPM: A visual, precise and multilevel metamodeling framework for describing mathematical domains and UML, *Springer-Verlag*, 2003.

[2] P.P. da Silva and N.W. Paton. User Interface Modelling with UML, 2000.

[3] M. Iseger. Domain-specific modeling for generative software development, 2005, `http://www.itarchitect.co.uk/articles/display.asp?id=161`

[1] This work was partially supported by the SENSORIA European project(IST-3-016004)